

Wie sich "Cross Site Scripting" auf Ihr Unternehmen auswirkt

Andreas Wiegenstein
Version 1.1- 2007-07-26

Dieses Whitepaper gibt eine generelle Übersicht über die Risiken von Cross Site Scripting Angriffen.

Funktionsweise von Cross Site Scripting Angriffen

Eine Cross Site Scripting (XSS) Schwachstelle entsteht, wenn eine Webanwendung Benutzereingaben anzeigt, ohne zuvor HTML Tags korrekt herauszufiltern.

Beispiel 1

Die meisten Websites bieten eine Suchfunktion an. Üblicherweise werden dabei die Suchergebnisse zusammen mit den Suchbegriffen angezeigt. Wenn Sie z.B. nach `XSS` suchen und `XSS` erscheint anschließend als Suchbegriff fett gedruckt, dann haben Sie einen XSS Defekt gefunden.

Beispiel 2

Eine Online Bewerbung enthält ein Feld "Motivation". Wenn ein boshafter Benutzer hier Tags eingibt, so kann dadurch die Seite manipuliert werden, die ein Recruiter öffnet, um die Bewerbung zu lesen.

Ein XSS Angriff funktioniert, indem der Inhalt einer Web basierten Anwendung verändert und dann *jemand anderes* dazu gebracht wird, diese Seite zu öffnen. Im ersten Beispiel müsste also ein Angreifer einen speziellen Link konstruieren und diesen dann in einem Forum veröffentlichen oder per E-Mail versenden. Im zweiten Beispiel funktioniert der Angriff allerdings "per Design", denn die Eingaben werden in einer Datenbank gespeichert und automatisch an das Opfer übermittelt.

Schadenspotential

Durch einen XSS Defekt kann beliebiger (JavaScript) Code im Browser des Opfers ausgeführt werden. Eine kleine Teilmenge von potentiellen Schäden zeigt folgende Übersicht:

- Die Session eines (angemeldeten) Benutzers kann gestohlen und in dessen Namen vom Angreifer weitergeführt werden
- Manipulation von Dateien auf sämtlichen Shares, auf die das Opfers legitimen Zugriff hat
- Ausspionieren sämtlicher Tastatureingaben in einer Webanwendung
- Diebstahl von Dateien aus sämtlichen Shares, auf die das Opfers legitimen Zugriff hat
- Abscannen des Intranets (in dem sich das Opfer befindet) nach weiteren Schwachstellen
- Gezieltes Angreifen von Systemen, die ein Opfer per Browser (im Intranet) erreichen kann
- Ausführung von "brute force" Angriffen gegen Anmeldeseiten mittels des "gekaperten" Browsers

Wie hier leicht zu erkennen kann, stellen XSS Schwachstellen gravierende *Compliance-Verstöße* dar, da sie sich auf Datenschutz und Nachvollziehbarkeit auswirken. Unternehmen können gesetzlichen

Vorschriften wie SOX und Datenschutz nicht genügen, solange ihre Geschäftsanwendungen XSS Defekte enthalten.

Kritische Faktoren

Die besondere Gefahr von XSS Schwachstellen liegt darin, dass diese von Angreifern *leicht erkannt* und *leicht ausgenutzt* werden können. Sie sind außerdem völlig plattformunabhängig. Im Gegensatz zu anderen Angriffen, die nur auf einem bestimmten Betriebssystem oder bei einer speziellen Datenbankversion funktionieren, läuft JavaScript Code in jedem Browser auf jedem System.

Zudem gibt es viele technische Varianten von XSS Schwachstellen, da HTML eine Sprache ist und man in jeder Sprache viele Möglichkeiten hat, die gleiche Information darzustellen.

Unsere Erfahrung aus zahlreichen Sicherheitsanalysen und Trainings zeigt außerdem, dass selbst erfahrene Entwickler die das Problem verstanden haben, zumeist nicht in der Lage sind Webanwendungen gänzlich ohne XSS Defekte zu schreiben.

Fazit

Eine einzige XSS Schwachstelle in einer beliebigen Webanwendung kann einem Unternehmen erheblichen Schaden zufügen. Ein Angriff trifft zwar zunächst nur einen Benutzer, kann sich aber von dessen Browser aus leicht auf andere Systeme ausweiten.

Stand heute haben mehr als 80% aller Webapplikationen Cross Site Scripting Probleme. Aufgrund dieser Fülle an Möglichkeiten werden Hacker Ihre Fertigkeiten entsprechend anpassen.

Es liegt in der Verantwortung eines jeden Unternehmens, die Privatsphäre und die Daten seiner Kunden und Mitarbeiter durch regelmäßige Sicherheitstests und Best Practices zu schützen.

Jemand wird die Schwachstellen in Ihren Anwendungen finden. Tun Sie es, bevor es ein Hacker tut.

Weitergehende Informationen

Online Informationen

- [1] The Web Hacking Incidents Database
<http://www.webappsec.org/projects/whid/>
- [2] The Cross Site Scripting Threat
http://www.virtualforge.de/cross_site_scripting_threat.php
- [3] Cross Site Scripting animated learning material
<http://www.virtualforge.de/vmovie.php>

Literatur

- [4] Gary McGraw, Software Security: Building Security In, Addison-Wesley Professional, 2006
- [5] Michael Howard, David LeBlanc, and John Viega, 19 Deadly Sins of Software Security – Programming Flaws and How to Fix Them, Osborne McGraw-Hill, 2005